타입 정보를 이용한 문서 매칭 기법 설계

○
고승규*, 임순범**, 최윤철*

*연세대학교 컴퓨터과학과, **숙명여자대학교 멀티미디어학과

The Design of Document Matching Method using Type Information

Seung-Kyu Ko*, Soon-Bum Lim**, Yoon-Chul Choy*
*Dept. of Computer Science, Yonsei University,
**Dept. of Multimedia Science, Sookmyung Women's University
E-mail: *{pitta, ycchoy}.rainbow.yonsei.ac.kr, **sblim@sookmyung.ac.kr

요 약

XML 문서가 널리 사용됨에 따라 XML 문서 간의 통합이나 변환의 필요성이 증가하고 있다. 이러한 변환이나 통합을 위해서는 미디에이터나 웨어하우스와 같은 방법이 이용될 수 있다. 그런데 어떤 방법을 이용하더라도 질의어와 지역 DTD 간의 매칭이나 지역 DTD 간의 매칭은 필수적이다. 따라서 매칭은 변환이나 통합을 위한 기본적인 기술이라고 볼 수 있다. 이와 같은 매칭 관련 연구는 관계형 데이터베이스 분야에서 많이 진행되었으며, 근래에 SGML/XML 분야에서도 연구가 진행되고 있다. 이중 SGML/XML 문서와 관련된 매칭은 주로 엘리먼트 이름과 구조 정보만을 이용하고 있으며, 특히 구조 정보를 이용할 경우에 잘못된 매칭을 유발시킬 수 있다. 이는 구조 정보가 의미 정보를 적절히 표현하지 못하고 있기 때문이다. 따라서 본 논문에서는 XML 문서에서 추출 가능한 타입 패턴을 정의하고, 이를 이용한 매칭 기법을 제안한다. 이 기법은 구조 정보를 이용하는 기존의 매칭 기법보다 좀 더 명확하고, 정확한 매칭이 가능하다. 또한 이는 타입 정보를 사용할 수 없는 DTD 기반의 XML 문서에서의 매칭 정확도를 높여줄 수 있을 뿐만 아니라 타입에 기본적인 의미 정보도 반영되므로 의미 기반 웹에 사용될 수 있다.

1. 서론

W3C에 의해 웹 표준 문서로 제정된 XML은 상호 운영성, 스타일과의 독립성, 확장성 등의 장점으로 인해 널리 사용되고 있다. 또한 최근에 발표된 웹 관련 문서 표준들은 대부분 XML에 기반하고 있다. 예를 들어 전자 상거래 표준인 ebXML, xCBL(Common Business Library), cXML(Commerce XML)과 문서 표준인 TEI(Text Encoding Initiative), ISO12083, DocBook, 그리고 전자책 표준인 EBKS(EBook of Korea Standard), JepaX, OEB PS(Open eBook Publication Structure) 등은 전부 XML에 기반하여 정의되었다. 이와 같이 XML이 널리 사용됨에 따라 기업 간의 XML 문서를 교환하거나 공유하는 경우가 발생하고, 심지어 기업 간의 합병이나 협업 작업을 위하여 XML 문서 간의 변환이나 통합의 필요성이 증가하고 있다.

이러한 변환이나 통합을 위한 방법은 두 가지로 나눌 수 있다. 첫번째는 각각의 문서는 그대로 있고, 중간에 미디에이터를 이용하는 것이고, 두 번째는 실제로 각 지역 문서를 통합하는 웨어하우스를 이용하는 것이다. 두 방법에서 공통적으로 중요한 것은 매칭이다. 즉, 첫번째 방법에서는 질의어와 지역 DTD 간의 매칭이 필요하며, 두 번째 방법에서는 지역 DTD 간의 매칭이 필요하다. 아직까지는 이러한 매칭 작업은 주로 수작업으로 이루어지고 있다. 그러나 점차로 문서의 크기가 커짐에 따라 이러한 수작업으로 매칭을 수행하기가 점점 어려워지고 있다. 따라서 매칭을 자동으로 수행하거나 수작업을 도와주는 방법들이 제안되고 있다. 그런데 제안 방법들은 구조 정보를 해석하는데 잘못된 매칭을 유발할 수 있다. 이러한 잘못된 매칭은 인식하고 정정하기 위해서는 많은 비용이 발생한다.

이에 본 논문에서는 XML 문서에서 추출가능한 타입을 정의하고 이를 위한 타입 패턴을 정의하였다. 그리고 타입을 이용한 매칭 기법을 제안한다. 이 기법은 기존의 XML DTD에서 제공하지 못한 타입 정보 뿐 아니라 기본적인 의미 정보까지도 이용 가능하여 효과적이고 정확한 매칭이 가능하다.

본 논문의 구성은 다음과 같다. 2절에서는 문서 통합과 관련된 연구에 대해 살펴보고, 3절에서는 매핑을 정의하기 위한 타입 패턴을 소개한다. 4절에서는 제안된 패턴을 이용한 매칭 기법에 대해 설명한다. 그리고 5절에서는 결론 및 향후 연구에 대해 기술한다.

2. 관련 연구

문서를 변환하거나 통합할 경우에는 각 엘리먼트를 대응하는 다른 엘리먼트로 매칭시켜야 한다. 그런데 똑같은 정보도 문서마다 다르게 표현될 수 있고, 이러한 점이 자동적인 매칭을 수행하기 어렵게 한다. 예를 들어 저자 정보는 "author", "writer" 등으로 표현될 수 있으며, 어떤 문서에서는 인물 정보를 이름, 주소, 전화번호로 세분하는데, 다른 문서에서는 인물 정보를 하나로 표현할 수 있다. 이와 같이 정보의 표현 방법의 차이로 인해 발생할 수 있는 문제점을 스키마 충돌이라고 한다. 본 절에서는 스키마 충돌과 기존의 매칭 기법에 대해 간략히 살펴본다.

2.1 스키마 충돌

XML 문서에서 발생할 수 있는 충돌은 다섯 가지로 분류할 수 있다.

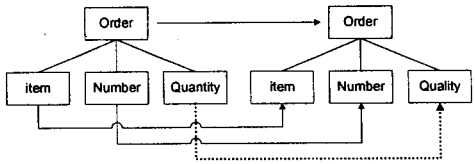
- 엘리먼트
  한 엘리먼트로 표현된 정보가 다른 문서에서는 여러 엘리먼트로 표현되거나 없는 경우가 이에 해당한다. 기존의 매칭 방법에서는 정확한 대응을 위하여 동의어 사전이나 약어 사전 등을 이용하고 있다.
- 구조
  하나의 논리적인 단위가 다른 문서에서는 여러 논리 단위나 다른 논리 단위와 섞여서 표현되는 경우가 이에 해당한다. 즉 논리 단위의 구조가 다른 경우가 이에 해당한다. 이와 같은 경우는 자동적인 매칭이 어렵고, 대부분의 시스템에서는 구조정보를 확대 해석하여 잘못된 엘리먼트로 매치시키고 있다.
- 속성
  속성으로 표현되는 정보 간에 발생하는 문제로 엘리먼트 문제와 유사하다.
- 엘리먼트-속성
  같은 정보가 엘리먼트로 표현될 수도 있고, 속성으로도 표현될 수 있는 경우가 이에 해당한다.
- 기타
  - 단위: 한 문서에서는 미터를 이용하고 다른 곳에서는 마일을 이용하는 경우에 둘 간의 적절한 변환이 필요하다.
  - 값: 내용이 표현하는 경우가 서로 차이가 나는 경우를 의미한다. 예를 들어 "최근의 내용"이라는 것은 그 문서가 생성된 날짜에 의존적이다. 이 경우는 자동으로 처리하기 매우 어려운 부분이다.

2.2 기존의 매칭 기법

기존의 매칭 방법은 개별 문서 정보 이용 여부와 구조 정보 이용하는지 여부에 따라 다음과 같이 분류할 수 있다.

- 스키마-인스턴스
  매칭 기법은 스키마 정보만을 이용하는지와 개별 문서 정보를 이용하는지에 따라 구별할 수 있다. 기존의 대부분의 매칭 기법은 스키마 정보만을 이용하고 있으나 개별 문서로부터 추출된 정보를 이용할 수도 있다. 예를 들어 엘리먼트를 대표하는 키워드는 실제 문서 내용으로부터 추출할 수 있다. 그리고 이 정보는 각 엘리먼트를 비교하는데 사용될 수 있다.
- 엘리먼트-구조
  매칭을 정의하는데 엘리먼트 정보만을 이용할 수도 있지만 구조 정보를 이용하여 수행할 수도 있다. 예를 들어 [그림 1]에서는 "item"과 "Number"가 매치가 되면, 이 대응(sibling) 정보를 이용하여 "Quantity"와 "Quality"가 매칭될 수 있다고 간주한다. 이와 같이 구조를 이용하면 매칭의 자동성은 높아지나 매칭의 정확도가 떨어지며, 이를 정정하기 위한 비용은 매우 크다.

대표적인 매칭 시스템으로는 1998년 텔아비브 대학에서 SGML 문서와 객체 지향 데이터베이스 간의 변환 기법으로 TranScm[7]이 제안되었다. 이 기법은 엘리먼트 정보만을 이용하고, 루트에서 단말



[그림 1] 구조 정보를 이용한 매칭 예

엘리먼트로 매칭하는 하향식 방법을 이용하고 있다. 2000년 워싱턴 대학에서 개발한 LSD[1][2]는 XML 문서 간의 통합을 위하여 제안되었으며 엘리먼트 정보뿐 아니라 구조 정보도 이용한다. 이 방법은 사용자가 초기에 매치를 하고, 이를 학습하여 매칭을 시키는 방법이다. 2001년 마이크로소프트에서 개발한 CUPID[3]는 XML 문서외에 관계형 데이터베이스 간의 변환을 위해 개발되었다. 이 기법은 특히 엘리먼트 매칭 시 이름 매칭을 위하여 WORDNET을 이용하고 있다. TranScm은 초기 시스템으로 매칭율이 떨어지며, 다른 방법들은 구조 정보와 동의어, 약어 사전 등을 이용하여 매칭율을 높였으나 [그림 1]과 같은 잘못된 매칭을 발생시킬 수 있다.

3. 타입 패턴

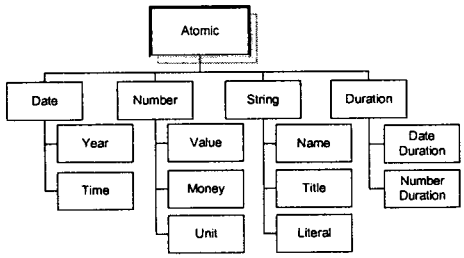
XML DTD에서는 문자열 타입만이 이용 가능하므로 실제 매핑을 정의할 때에 명확한 매핑을 정의하기 어렵다. 이에 본 논문에서는 XML 문서에서 추출 가능한 타입을 분석하고, 이를 위한 타입 패턴을 정의하였다. 이러한 타입 정보는 각 엘리먼트 간의 명확한 매칭을 가능케 한다.

XML 문서에서 발생 가능한 타입은 기본 타입 패턴과 복합 타입 패턴으로 구별할 수 있다. 기본 타입 패턴은 다시 숫자 타입과 문자열 타입으로 구별할 수 있으며, 집합 타입은 동종 타입과 이종 타입으로 나눌 수 있다.

3.1 기본 타입 패턴 (atomic type pattern)

기본 타입 패턴은 실제 내용을 포함하는 엘리먼트의 타입을 표현할 수 있는 타입을 의미한다. XML 문서를 살펴보면 일반적인 내용은 대부분 <p> 엘리먼트로 표현되며, 이외의 다른 엘리먼트로 표현되는 경우에는 나름대로의 논리적인 단위를 표현한다. 이러한 논리적인 단위는 단말 노드(엘리먼트)와 비단말 노드로 구별할 수 있으며, 각각에 따라 기본 타입과 복합 타입으로 구별할 수 있다. 기본 타입은 [그림 2]와 같이 크게 숫자형, 문자형, 기간형, 날짜형으로 구별된다.

① 숫자 타입
이 타입의 예로는 장이나 절 등의 번호, 금액, 통계 값 등이 해당한



[그림 2] Atomic 타입 패턴

다. 이 때 금액을 나타내는 경우에는 콤마를 이용하여 표현할 수도 있으며, 단위를 포함하는 경우도 있을 수 있다. 이 타입은 금액 타입과 단위 타입, 그리고 값 타입으로 구별된다. 각 타입의 패턴은 다음과 같다.

Money

If content pattern is in

123,234원, 123,345$, 213,231Won, …

Unit

If content pattern is in

12345 cm, 12313 m^2, 12335 ℃, ….

우편번호, 전화 번호 등도 이에 해당한다. 실제 매칭에 이용할 경우에는 우편번호, 전화번호 등과 같은 수준까지 추출한다.

Value

If content includes the numeric values, and its size is more than a half of it.

숫자 타입은 다른 타입과 달리 범위를 지닐 수 있다. 따라서 각 문서들에서 타입을 추출할 때 범위도 동시에 추출하여 이용한다.

② 날자 타입

날자 타입은 년도와 시간으로 구별가능하며, 각 타입의 패턴은 다음과 같다.

Year

If content pattern is in

YYYY-MM-DD, YYYY년 MM 월 MM일, YY-MM-DD, YYMMDD, MM-DD-YYYY, MMDDYYYY, DD Jan. YY(YY), …

Time

If content pattern is in

HH:MM:SS, HH:MM, HH시 MM분 SS초, …

③ 문자열 타입

문자열 타입은 이름과 문자(literal)로 구별가능하다.

Name

If the size of content value is consist of three in a biography dictionary.

이름의 종류로는 사람 이름, 회사 이름, 절 이름 등 여러 가지가 있지만 자동으로 이러한 것을 인식하기는 어렵다. 본 논문에서는 이름과 고유명사만을 분류하고, 이를 위하여 다음과 같은 방법을 이용한다.

1. 엘리먼트 내용의 길이가 한글의 경우에 2-4 자이고, 영어의 경우에 20자 이하이면 인명의 후보로 간주한다. 인명인지 여부는 다음이나 야후의 사람 찾기 기능을 이용한다.
2. 엘리먼트 내용의 길이가 한글의 경우에 10자 이하이고, 영어의 경우에 30자 이하이면 회사 이름 등의 고유 명사의 후보로 간주한다. 이 경우에도 검색 사이트에서 정확한 매칭(exact matching) 검색을 시도하여 검색 결과가 있으면 이름 패턴으로 간주한다.

Literal

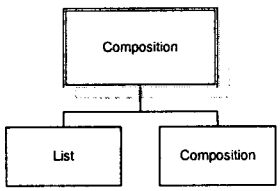
이름이 아닌 정보를 의미하며, 주소나 기타의 내용이 이에 해당한다.

④ *기간 타입*

이 타입은 기간을 의미하며, 시간이나 날짜 등의 기간이 이에 속할 수 있다.

3.2 복합 타입 패턴 (atomic type pattern)

실제 내용을 지닌 단말 엘리먼트들을 논리적인 단위로 묶어 주는 것이 비단말 엘리먼트들이다. 이러한 비 단말 엘리먼트들은 [그림 3]과 같이 두 가지로 분류할 수 있다.



[그림 3] Composite 타입 패턴

List 타입은 하위 엘리먼트들이 동일한 타입으로 구성되어 있는 경우를 의미한다. 예를 들어 텍스트 정보만을 지니고 있는 <p> 엘리먼트들의 모임이나 <author> 정보의 모임인 <author_grp> 등이 이 패턴에 해당한다. Composition 타입은 하위 엘리먼트의 타입이 동일하지 않을 경우가 이에 해당한다. 대부분의 비단말 엘리먼트들이 이에 해당한다.

4. 매칭 기법

기존의 매칭 기법들은 대부분 엘리먼트의 이름과 구조 정보에 기반하여 수행된다. 그런데 엘리먼트 이름 이외에 구조 정보를 이용하면 [그림 1]과 같이 잘못된 매칭이 발생할 수 있다. 본 논문에서는 타입 패턴 정보를 이용하여 잘못된 매칭을 최소화시키는 매칭 기법을 제안한다. 본 논문에서 제안하는 매칭 기법은 먼저 개별 문서의 구조 정보를 이용하여 스키마와 타입 정보를 추출한다. 그리고 매칭 시에는 엘리먼트의 이름, 타입 정보, 그리고 구조 정보에 기반한 유사도를 이용한다.

4.1 매칭 기법

본 매칭 기법에서는 Dataguide[4]와 Maximum Tree[6]와 같이 개별 문서로부터 스키마를 생성한다. 그리고 생성된 스카마를 통합하는 하나의 통합 스키마를 생성한다. 이 때 매칭이 발생하며, 매칭이 안되는 엘리먼트 간의 관계는 정보량의 대소를 표시하여 사용자가 손쉽게 매칭 작업을 수행할 수 있도록 한다. 그리고 문서 변환을 위해서는 통합된 스키마에서 정보량의 대소를 이용하여 매칭을 시도한다.

XML 문서의 매칭은 먼저 기준 스키마를 결정하고, 이를 기준으로 그 하위의 자식 노드들을 비교한다. 순서는 먼저 루트가 같다고 가정을 하고, 그 하위로 가는 하향식 방법을 사용한다. 비교되는 노드들이 매치가 안될 경우가 발생할 수 있는데, 이 경우에는 자식 노드들을 매치시키고 그 정보를 이용하여 매치하게 된다. 노드들을 비교할 때에는 유사도 함수를 기준으로 하고, 여러 노드들이 동시에 매치가 될 경우에는 노드 선택 기준을 이용한다. 기준 스키마를 결정하는 기준은 다음과 같다.

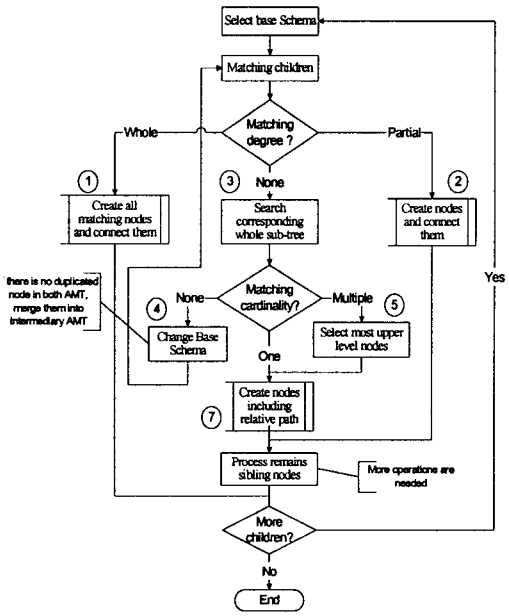
기준 스키마 결정 기준

1. 비교 레벨에서 매치되지 않은 노드의 수가 많은 스키마
2. 비교 하위의 레벨에서 매치되는 않는 노드의 수가 많은 스키마

그리고 매칭 시 여러 노드들이 매치될 경우에는 다음과 같은 기준에 따라 매칭되는 노드를 선택한다.

노드 선택 기준

1. 유사도 값이 높은 노드
2. 매칭된 노드의 수가 많은 노드
3. 매칭된 노드의 레벨이 높은 경우



[그림 4] Matching 흐름도

그리고 매칭이 되지 않는 노드들은 자식 엘리먼트들을 이용하여 매칭을 시키는데 이때 부모 선택 규칙이 이용된다.

1. 자식 노드들이 전부 매치되면, 그 부모도 매치시킨다.
2. 자식 노드들이 부분적으로 매치가 되면, 좀 더 넓은 의미를 갖는 노드를 상위로, 그렇지 않은 노드를 하위로 매치시킨다.

구체적인 매칭 과정은 [그림 4]와 같다.

4.2 유사도 함수

기존의 XML 문서 매칭 기법과는 달리 본 논문에서는 타입 정보도 함께 고려하여 매칭을 수행한다. 본 논문에서는 매칭을 위하여 다음과 같은 유사도 함수를 이용한다.

$$Similarity = Sim_n * w_n + Sim_t * w_t + Sim_s * w_s$$

엘리먼트 이름과 관련된 유사도(Sim_n)는 동의어 사전과 약어 사전 등을 이용하여 구할 수 있고, 타입과 관련된 유사도(Sim_t)는 타입 패턴을 이용하여 구할 수 있다. 그리고 구조와 관련된 유사도(Sim_s)는 자식 노드와 형제 노드 간의 매칭된 결과를 이용하여 구할 수 있다. 그리고 각각의 유사도에 가중치를 곱하여 유사도를 구하게 된다. 이 중에서 이름과 관련된 정보는 의미 정보를 반영하고 있으며, 타입 정보는 부분적으로 의미 정보를 반영하고 있으므로 가중치 간의 다음과 같은 관계가 성립한다.

$$w_n > w_t >= w_s$$

타입을 고려하지 않고, 제안된 매칭 기법을 이용하여 표준 전자책 문서 변환에 적용해본 결과, [5]와 유사한 결과를 얻었다. 제안된 유사도 함수를 이용하면 좀 더 효과적인 매칭 결과를 얻을 수 있을 것으로 예상된다.

5. 결론

본 논문에서는 문서 변환이나 통합 등에 사용되는 핵심 기술인 문서 매칭 기법에 대해 살펴보았다. 제안 기법은 XML 문서에서 타입 정보를 추출하여 매칭에 사용함으로써 다른 매칭 기법보다 정확한 매칭을 제공할 수 있을 것으로 예상된다. 현재 제안 기법은 전자책 표준 문서의 변환에 적용 중에 있으며, 유사도의 가중치를 변화시켜 가면서 유사도 함수에서 엘리먼트의 이름과 타입 그리고 구조 정보가 갖는 중요도를 실험 중에 있다.

6. 참고문헌

[1] AnHai Doan, Pedro Domingos, Alon Y. Halevy, "Reconciling Schemas of Disparate Data Sources: A Machine-Learning Approach." SIGMOD Conference 2001.

[2] AnHai Doan, Pedro Domingos, Alon Y. Levy, "Learning Source Description for Data Integration," In Proc. of WebDB, pp. 81-86, 2000.

[3] Jayant Madhavan, Philip A. Bernstein, Erhard Rahm, "Generic Schema Matching with Cupid," In Proc. of 27th International Conference on Very Large Data Bases, pp. 49-58, 2001.

[4] R. Goldman and J. Widom, "DataGuides: Enabling query formulation and optimization in semistructured databases," In Proc. of the 23th International Conference on Very Large Data Bases, Athens, Greece, pp. 436-445, August 1997

[5] Seung-Kyu Ko, Myong-Soo Kang, Won-Sung Sohn, Soon-Bum Lim, Yoon-Chul Choy, "Converison of eBook Documents based on Mapping Relations,", ECDL2002, Italy, Sep 2002, LNCS 2458, pp. 32-46, Springer.

[6] Seung-Kyu Ko, Yoon-Chul choy, "A Structured Documents Retrieval Method supporting Attribute-based Structure Information", Proc. 17th ACM Symposium on Applied Computing, March. 2002.

[7] Tova Milo, Sagit Zohar, Using Schema Matching to Simplify Heterogeneous Data Translation, Proc. of VLDB, pp. 122-133, 1998